IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
MARSHALL DIVISION

Rembrandt Patent Innovations, LLC and Rembrandt Secure Computing, LP,	§	
	§	Civil Action No. 2:14-cv-00015
Plaintiffs,	§	
v.	§	
	§	JURY TRIAL REQUESTED
Apple Inc.,	§	
	§	
Defendant.	§	

DEFENDANT APPLE INC.'S RESPONSIVE CLAIM CONSTRUCTION BRIEF

TABLE OF CONTENTS

TABLE OF AUTHORITIES

Page(s)

Cases

Statutes

Other Authorities

I. INTRODUCTION

Defendant Apple Inc. ("Apple") proposes the following constructions for the disputed claims terms of U.S. Patent 6,185,678 (the "'678 Patent"). As discussed further, Apple's constructions come from the intrinsic language of the '678 Patent and are consistent with a person of ordinary skill's understanding of the invention at the time the '678 Patent was filed.

II. THE '678 PATENT

Filed on October 2, 1998, the '678 Patent "relates to an architecture for initializing a computer system and more particularly to a secure bootstrap process and automated recovery procedure." '678 Patent at col. 1:23–25.

The "bootstrap process," also known as the boot process or booting, refers to all of the steps between powering on a computer and loading the operating system that allows the user to interface with the computer. The boot process is managed by the computer's BIOS (Basic Input/Output System). The boot process begins with a series of diagnostic checks commonly referred to as POST (Power On Self-Test). The BIOS also initializes boot components needed for the computer to function, including boot devices, such as hard drives, that can load an operating system. Once the initialization process is complete, the BIOS passes control to the operating system. (Declaration of Dale Buscaino in Support of Apple Inc.'s Responsive Claim Construction Brief ("Buscaino Decl.") ¶ 3.) *See also* '678 Patent at col. 8:12–31.

The '678 Patent seeks to improve the boot process by verifying that the BIOS, the boot components, and the operating system are in the condition expected by the system before they are allowed to operate, and, if a verification check fails, automatically recovering from the failure by obtaining a replacement of necessary software or data. "The present invention does this by constructing a chain of integrity checks, beginning at power-on and continuing until the final transfer of control from the bootstrap components to the operating system itself. . . . Once an integrity failure is detected, the invention uses a secure protocol to inform a trusted repository that a failure has occurred and to obtain a valid replacement component." '678 Patent at col.

4:40–51. "The trusted repository can either be an expansion ROM board . . . or it can be a network host[.]" '678 Patent at col. 10:44–46.

Comparing Figure 1a with Figure 2a of the '678 Patent illustrates the differences the inventors believed distinguished their invention from the prior art. Figure 1a is a functional diagram of the prior art. Figure 2a "is a functional diagram of the functional layers of the AEGIS embodiment of the bootstrap process of the current invention." '678 Patent at col. 5:42–44. As shown in these figures, the primary structural differences between the '678 Patent and the illustrated prior art are the addition of the AEGIS ROM (#256) and the network host (#254). The AEGIS ROM manages the automated verification and recovery process discussed in the Patent. The network host is one potential "trusted repository" that can store replacement boot components. '678 Patent at col. 10:44–46.

Figure 2b is a flowchart showing the sequence of verification checks that are performed during the bootstrap process in the AEGIS embodiment of the '678 Patent. The below version of Figure 2b is annotated with colors that show the flow of the automated verification and recovery process.

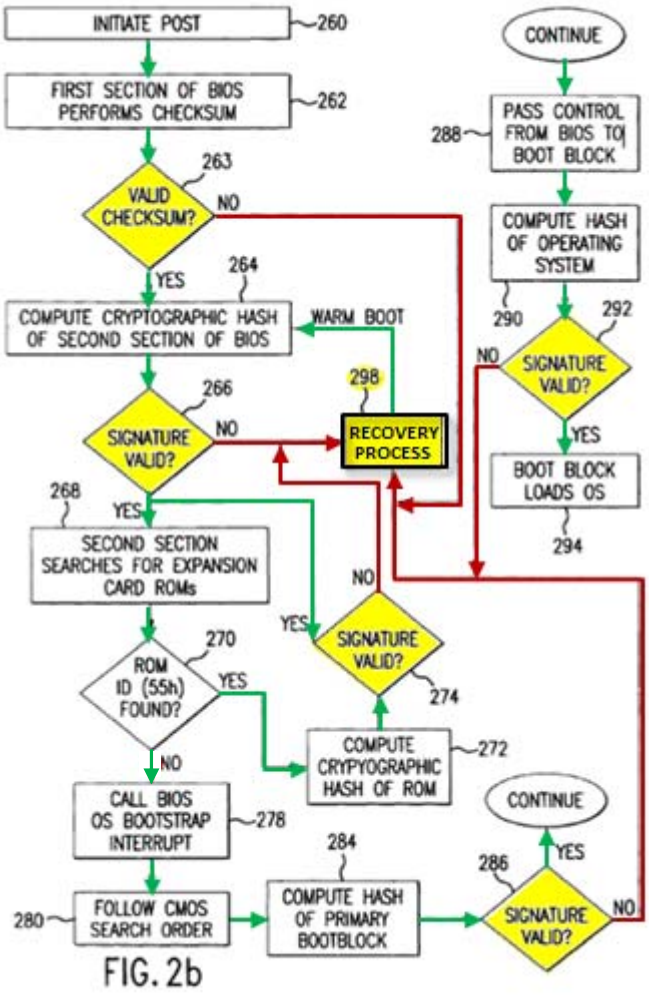

FIG. 2b

When an integrity check fails, the recovery process is entered (step 298). If a component fails its integrity check, its recovery is managed by the AEGIS ROM. If the failed component is a portion of the BIOS, the failure is recovered directly from the AEGIS ROM, by using a copy of the BIOS software on the AEGIS ROM instead of the software on the ROM BIOS – what the Patent calls "shadowing the failed component." '678 Patent at col. 10:56–61. The BIOS is not repaired. Instead, its function is performed by the AEGIS ROM each time the system starts. "A failure beyond BIOS 112 causes the system to boot into a recovery kernel contained on AEGIS ROM 256. The recovery kernel contacts a 'trusted' host through a secure protocol . . . to recover a verified copy of the failed component. The failed component is then shadowed or repaired, if

possible, and the system is restarted." '678 Patent at col. 10:61–67. The BIOS and boot component verification and recovery "process occurs without user intervention." '678 Patent at col. 6:25.

III. LEGAL STANDARD

A court should give each claim term the "meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention." *Phillips v. AWH Corp.*, 415 F.3d 1303, 1313 (Fed. Cir. 2005) (en banc). To determine the proper meaning of a disputed term in a patent claim, courts looks first to the intrinsic evidence, namely the claim language itself, the specification, and the prosecution history. *Id.* at 1315, 1317. In *Phillips*, the Federal Circuit described the evidence that a court should consider in interpreting claims. The most reliable form of evidence is the patent (the claims and specification) and its prosecution history, because this "intrinsic evidence" provides "evidence of how the PTO and the inventor understood the patent." *Id.* at 1317. The specification is particularly important and "is always highly relevant to the claim construction analysis. Usually, it is dispositive; it is the single best guide to the meaning of a disputed term." *Id.* at 1315 (citations omitted); *see also Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582 (Fed. Cir. 1996). The specification may define terms explicitly or by implication. *IrdetoAccess, Inc. v. Echostar Satellite Corp.*, 383 F.3d 1295, 1300 (Fed. Cir. 2004) ("Even when guidance is not provided in explicit definitional format, the specification may define claim terms by implication such that the meaning may be found in or ascertained by a reading of the patent documents.") (internal quotation marks omitted); *Phillips*, 415 F.3d at 1321. Sometimes, a patentee will act as his own lexicographer, coining terms with unique meanings in the context of the patent. *Edwards Lifesciences LLC v. Cook Inc.*, 582 F.3d 1322, 1329 (Fed. Cir. 2009). Such terms "are best understood by reference to the specification." *Intervet Inc. v. Merial Ltd.*, 617 F.3d 1282, 1287 (Fed. Cir. 2010).

A claim term should not be read beyond the disclosed embodiments where such a reading is contrary to "the written description['s] . . . guidance as to the meaning of the claims." *SciMed Life Sys., Inc. v. Advanced Cardiovascular Sys., Inc.*, 242 F.3d 1337, 1344 (Fed. Cir. 2001)

(where catheters in the art were either coaxial or side-by-side, and specification discussed only coaxial catheter, this "lead[s] to the inescapable conclusion" that only coaxial catheter is claimed); *see also Kinetic Concepts, Inc. v. Blue Sky Med. Group, Inc.*, 554 F.3d 1010, 1018–19 (Fed. Cir. 2009); *Wang Labs., Inc. v. Am. Online, Inc.*, 197 F.3d 1377, 1381, 1383 (Fed. Cir. 1999).

IV. ARGUMENT

A. "verifying the integrity"

Terms & Claims	Apple's Construction	Rembrandt's Construction[1]
"verifying the integrity" (Claims 1 and 4)	"confirming the expected condition"	"checking the integrity using a cryptographic hash"

Rembrandt's proposed construction narrows the scope of "verifying the integrity" by simply substituting the word "checking" for "verifying" and then importing the limitation that the verification be done "using a cryptographic hash." First, "checking" is not an appropriate substitution for "verifying." To the extent Rembrandt is simply proposing a synonym for "verifying," using the word "checking" is no more helpful to the jury than the word "verifying" and is thus unnecessary. To the extent Rembrandt believes "checking" means something more illustrative than "verifying," Rembrandt has failed to explain how or why that is the case.

Second, the intrinsic evidence does not support construing this term so narrowly as to require the use of a cryptographic hash. The '678 Patent discloses two primary methods of integrity verification: (1) The integrity checks compare a computed cryptographic hash value with a stored digital signature associated with each component ('678 Patent at col. 6:14–16); and (2) the integrity check of the first portion of the BIOS using a ***non-cryptographic*** checksum, as shown in Figure 2b (steps 262 and 263). *See* Annotated Figure 2b, supra.

Rembrandt reads out this non-cryptographic verification method in its proposed construction. In doing so, Rembrandt ignores the non-cryptographic checksum performed on the

[1] "Rembrandt" refers collectively to Rembrandt Patent Innovations, LLC and Rembrandt Secure Computing, LP.

first portion of the BIOS, which is undoubtedly part of the integrity checking process. Indeed, while the '678 Patent attempted to innovate on prior art verification with respect to certain computer components, it disclosed only checksum verification of the first portion of the BIOS, as had been done in the prior art. The checksum was an integral part of the verification and recovery chain disclosed in the Patent, as shown in Figure 2b. The specification makes clear that the AEGIS embodiment is considered to perform an "integrity check" on both portions of the BIOS. *See* '678 Patent at col. 10:56–58 ("If the component that fails its integrity check is a portion of BIOS 112, then it must be recovered from AEGIS ROM 256."); *see also* '678 Patent at col. 4:40–41 ("The present invention . . . construct[s] a chain of integrity checks, beginning at power-on . . . .").

The doctrine of claim differentiation also counsels against Rembrandt's proposed construction. *Curtiss-Wright Flow Control Corp. v. Velan, Inc.*, 438 F.3d 1374, 1380 (Fed. Cir. 2006) ("an independent claim should not be construed as requiring a limitation added by a dependent claim"). Claim 6[2] of the '678 Patent, which is dependent on claim 4, discloses a method for verifying the integrity of a boot component using a cryptographic hash:

> (a) computing a cryptographic hash value for said boot component; and
>
> (b) comparing said cryptographic hash value with a digital signature associated with said boot component stored in a trusted memory location.

Rembrandt's proposed construction requiring the use of a cryptographic hash for verification would render that express limitation of dependent claim 6 superfluous.

Apple's proposed construction captures the common thread in the integrity verification methods discussed in the specification: comparing (1) a cryptographic hash value with a digital signature; and (2) a checksum with the sum of the bits in the component. In each case, a predetermined value is compared with a value calculated at the time of verification. This

[2] Rembrandt does not allege infringement of claim 6.

comparison confirms that the component being verified is in the expected condition, with the expected condition being represented by the predetermined comparison value.

B. "a trusted repository"

Terms & Claims	Apple's Construction	Rembrandt's Construction
"a trusted repository" (Claims 1, 3 and 7)	"an expansion ROM or network host that is assumed to operate as expected by the system and contains copies of the plurality of boot components"	"an expansion ROM or one or more network hosts that are assumed to operate as expected by the computer system"

The parties agree that the "trusted repository" includes either an expansion ROM or a network host that is "assumed to operate as expected." The dispute between the parties relates to whether a trusted repository may be more than one network host and whether it must contain copies of the boot components (*i.e.*, their software and/or configuration data).

Rembrandt argues that a trusted repository can be one or more network hosts based on a single reference to network hosts. However, nowhere does the Patent discuss a trusted repository residing on more than a single network host. Rembrandt argues that multiple network hosts are supported since "[t]he DHCP server 'provide[s] the name and server location of a bootstrap program to the client. . . .'" (ECF No. 76 at 13.) But, regardless of what services or servers are used to locate the repository, the Patent teaches that the repository only resides on a single network host. *See, e.g.*, '678 Patent at col. 10:44–46 (defining trusted repository to include "network host").

In addition, the intrinsic evidence demonstrates that the "trusted repository" is intended to contain copies of the boot components. The "trusted repository exists for recovery purposes." '678 Patent at col. 7:50–51. "Once an integrity failure is detected, the invention uses a secure protocol to inform a trusted repository that a failure has occurred and to obtain a valid replacement component." '678 Patent at col. 4:48–51. An express purpose of the trusted repository, then, is to be a repository for replacement boot components.

Rembrandt purposely misreads the specification to argue that Apple's construction would "exclude the network host" embodiments of a trusted repository. (ECF No. 76 at 14.) But Rembrandt's reading ignores other portions of the specification and misses the whole point of the invention. As Rembrandt points out, the specification states: "The trusted repository can either be an expansion ROM board, not shown, that contains verified copies of the required software or it can be network host 254." '678 Patent at col. 10:44–46. As Rembrandt acknowledges, this language expressly states that the "expansion ROM" version of the trusted repository contains "verified copies of the required software." But Rembrandt implies that because the "verified copies" phrase does not similarly follow "network host", the network host need not contain such copies. Rembrandt's reading is too mechanical. In the sentence at issue, the '678 Patent simply provides more detail about the functioning of the expansion ROM because the subsequent paragraphs are dedicated to explaining the functioning of the network host (and not the expansion ROM). *See* Ex. A at 10:47–67; 11:1–15. These subsequent paragraphs make clear that when a network host is the trusted repository, it contains verified copies of the components. *See, e.g.,* '678 Patent at col. 10:63–65 ("The recovery kernel contacts a 'trusted' host through a secure protocol, as discussed below, to recover a verified copy of the failed component."). Indeed, the word repository means "a receptacle or place where things are deposited [or] stored . . . ." (Declaration of Mark C. Scarsi in Support of Apple's Responsive Claim Construction Brief ("Scarsi Decl.") ¶ 2, Ex. A.) Rembrandt's proposed construction of "wherein integrity failures are recovered through said trusted repository," discussed below, demonstrates that Rembrandt, too, believes that the trusted repository contains replacement components.

C. "means for verifying the integrity of said boot components and said system BIOS wherein integrity failures are recovered through said trusted repository"

Terms & Claims	Apple's Construction	Rembrandt's Construction
"means for verifying the integrity of said boot components and said system BIOS wherein integrity failures are recovered through said trusted repository" (Claim 1)	**Function**: "verifying the integrity of the boot components and the system BIOS wherein replacement components are automatically obtained from a trusted repository"[3] **Structure**: Fig. 2a #254, 256; Figs. 2b-2d steps # 262, 263, 264, 266, 272, 274, 284, 286, 290, 292) Col. 10:44-46; Col. 10:47-67; Col. 11:1-8	**Function**: "verifying the integrity of said boot components and said system BIOS" **Structure:** "a verification function that performs a cryptographic hash of a layer (e.g., steps 264, 272, 284, and 290 in Figs. 2b-2d) and compares the result to the value obtained from a stored signature for the layer (e.g., steps 266, 274, 286, and 292 in Figs. 2b-2d), and equivalents thereof" **Wherein clause:** "wherein replacement components are obtained from a trusted repository"

The parties agree that at least part of the phrase at issue is a means-plus-function claim element governed by 35 U.S.C. § 112(6). However, Rembrandt argues that the underlined portion of the following phrase is not governed by § 112(6): "means for verifying the integrity of said boot components and said system BIOS wherein integrity failures are recovered through said trusted repository." But, because the "wherein clause" is part and parcel to the automated verification and recovery process disclosed in this claim element, it should not be separated from it. *See Griffin v. M. Bertina*, 285 F.3d 1029, 1033–34 (Fed. Cir. 2002) (wherein clause gives meaning and purposes to the manipulative steps).

The network host from which failed components are recovered (#254 in Figure 2a) is part of the means for recovery. Even if the recovery process is not considered a part of the means-

[3] As discussed below, Apple's construction of the function portion of this term is modified to crystallize for the Court the dispute between Apple and Rembrandt.

plus-function claim element, the AEGIS ROM (#256 in Figure 2a), which handles both verification and recovery must be included as part of the disclosed means. *See, e.g.,* '678 Patent at col. 10:49–52 ("BIOS 112 and AEGIS ROM 256 contain the verification code, and public key certificates. AEGIS ROM 256 also contains code that allows the secure recovery of any integrity failures found during the initial bootstrap.").

For the reasons discussed in the "verifying the integrity" section, *supra*, Rembrandt's description of the structure is inappropriate. Rembrandt restricts the structure of integrity verification to comparing a "cryptographic hash . . . to the value obtained from a stored signature," thereby ignoring the checksum integrity verification discussed in the specification. Relatedly, steps 262 and 263 in Figures 2b–2d, which relate to the BIOS checksum, should be included in the structure as Apple proposed. Here, again, Rembrandt's proposed construction violates the principles of claim differentiation without justification by reading "verifying the integrity" as narrowly as it is framed in claim 6, which specifically limits verifying the integrity to comparing a "cryptographic hash value with a digital signature."

In order to crystallize the parties dispute regarding the construction of the "wherein clause" (separate from the analysis of whether it should be part of the means-plus-function claim element), Apple has modified its construction of the clause so that it is identical to Rembrandt's except that it contains the word "automatically": "wherein replacement components are automatically obtained from a trusted repository."

The addition of the word "automatically" is appropriate because all of the intrinsic indicates that the '678 Patent was directed solely at automatic recovery, despite the fact that (or perhaps because) the inventors were well aware of recovery methods that required user intervention. Again, under the case law, a claim term should not be read beyond the sole disclosed embodiments where such a reading is contrary to "the written description['s] . . . guidance as to the meaning of the claims." *SciMed Life Sys., Inc. v. Advanced Cardiovascular Sys., Inc.*, 242 F.3d 1337, 1344 (Fed. Cir. 2001) (where catheters in the art were either coaxial or side-by-side, and specification discussed only coaxial catheter, this "lead[s] to the inescapable

conclusion" that only coaxial catheter is claimed); *see also Kinetic Concepts, Inc. v. Blue Sky Med. Grp., Inc.*, 554 F.3d 1010, 1018–19 (Fed. Cir. 2009); *Wang Labs., Inc. v. America Online, Inc.*, 197 F.3d 1377, 1381, 1383 (Fed. Cir. 1999). Here, the intrinsic evidence confirms that only automatic recovery is claimed—including through descriptions of "the present invention" as opposed to any particular embodiment. *See Modine Mfg. Co. v. U.S. Int'l Trade Commc'n*, 75 F.3d 1545, 1551 (Fed. Cir. 1996) ("[W]hen the preferred embodiment is described in the specification as the invention itself, the claims are not necessarily entitled to a scope broader than that embodiment."). This is a situation where the specification "consistently, and without exception" describes an embodiment with automatic recovery characteristics that are critical to the invention. *Curtiss-Wright*, 438 F.3d at 1379. In such situations, patentees may not extend their monopoly to matter lacking those characteristics. *Id.* (vacating construction broader than "overall context of . . . specification" permitted).

When describing the invention, the inventors repeatedly chose to include language regarding its automatic nature. For example, the inventors described the "Field of the Invention" as follows: "This invention relates to an architecture for initializing a computer system and more particularly to a secure bootstrap process and ***automated*** recovery procedure." '678 Patent at col. 1:23–25 (emphasis added). The first paragraph of the "Detailed Description of the Preferred Embodiments," which summarizes the verification and recovery process, ends with the following sentence: "This entire process occurs ***without user intervention***." '678 Patent at col. 6:25 (emphasis added); *see also* col.10:8 (same sentence).

The "Background of the Invention" further reflects that the inventors were solely focused on automated recovery. They criticized prior art recovery efforts that "required human interaction." '678 Patent at col. 3:43–44. "This is in contrast to the ***present invention*** which ***provides automatic recovery of all of the bootstrap components*** including ROM chips." '678 Patent at col. 3:57–59 (emphasis added). They noted that the benefits of this automatic approach included enhanced security and a reduction of "the Total Cost of Ownership (TCO) of a personal computer, through automatically detecting and repairing integrity failures, thereby permitting the

user to continue to work without the nuisance of a trouble call to support staff and the associated down time." '678 Patent at col. 4:60–65. Figures 2b–2d also show the automated nature of the verification and recovery process.

Rembrandt's assertion that the specification "also describes recovery more generally, without regard to whether it is performed automatically" is incorrect. Rembrandt attempts to support this assertion by citing to a description of the verification and recovery process of the AEGIS embodiment of the Patent. *See* '678 Patent at col. at 10:19–25. But, the verification and recovery process of the AEGIS embodiment is described throughout the specification as being an automatic process. *See, e.g.,* '678 Patent at col. 10:8 ("This entire process occurs without user intervention."); col. 11:1–8. Rembrandt's citation of one paragraph describing the AEGIS embodiment that does not happen to contain "automatic" or "without user intervention" does not change the fact that the remainder of the specification describes the AEGIS embodiment as automatic.

Even the language cited by Rembrandt standing alone makes it sufficiently clear that the recovery process is automatic:

> In each case, AEGIS attempts to recover from a trusted repository, step **298**, as discussed below. Should a trusted repository be unavailable after several attempts, then the client's further action depends on the security policy of the user. For instance, a user may choose to continue operation in a limited manner or may choose to halt operations altogether.

(*See* ECF No. 76 at 20.) The AEGIS system is in charge of the recovery process. The user only has the opportunity to act if the recovery process fails and AEGIS returns control to the user.

In short, the '678 Patent is clearly focused on automatic—and only automatic—verification and recovery. Thus, Apple's proposed construction is appropriate.

D. "boot component"

Terms & Claims	Apple's Construction	Rembrandt's Construction
"boot component" (Claims 1, 4 and 7)	"a module used by the system BIOS to initialize the computer system"	"one or more software layers used in computer initialization"

As Rembrandt acknowledges, the parties' dispute regarding the construction of "boot component" is two-fold. Rembrandt argues that a boot component must be only software, while Apple argues it must also include hardware and thus "module" is more appropriate than "one or more software layers." The parties also dispute whether a "boot component" must be "used by the system BIOS."

The '678 Patent defines boot components in two places: in the specification, by providing examples of boot components, and in claim 1, by explaining how boot components are used. The definitions in the Patent support Apple's proposed construction and should be given considerable weight. *See, e.g.*, *Irdeto Access, Inc. v. Echostar Satellite Corp.*, 383 F.3d 1295, 1300 (Fed. Cir. 2004) ("It is well-established that the patentee can act as his own lexicographer . . . .").

With respect to the parties' dispute as to whether a boot component must be software, the specification clearly identifies various pieces of hardware as boot components when it lists examples of boot components: "Designers of trusted systems often avoid this problem by including the boot components (including but not limited to the ***ROM BIOS*** (Basic Input Output System), any ***expansion card ROMs***, ***CMOS memory*** and ***NVRAM***, the boot sector and the operating system kernel). . . ." '678 Patent at col. 1:55–57 (emphasis added). The first four references in this list of boot components are clear references to hardware memories, not software.[4] For example, the "ROM BIOS" is described and shown in Figure 1c as a hardware element. '678 Patent at col. 6:49–51 ("Computer system 1 also includes a main memory 8, preferably random access memory (RAM) and a ROM BIOS 2, which stores the system BIOS."). The Patent describes the ROM BIOS as the "system BIOS" stored in Read Only Memory (ROM). *Id.* The hardware that stores the BIOS is listed as a boot component.

[4] "ROM" stands for "read only memory" and "RAM" stands for "random access memory." (Buscaino Decl. ¶ 5.)

In addition, a boot sector (the fifth reference listed) clearly has a physical component. A dictionary at the time defined "Boot sector" as "[t]he portion of a disk reserved for the bootstrap loader (the self starting portion) of an operation system." Sector is further defined as "A portion of the data storage area on a disk. . . ***Sectors are the smallest physical storage units on a disk***." (Scarsi Decl. ¶ 5, Ex. D (emphasis added).)

The specification uses the phrase "boot component" only once, in relation to the above-quoted list of boot components, which include hardware components. Moreover, as discussed below in the section regarding the term "coupled to," claim 1 requires "boot components" to be coupled to a system bus. Software alone is not "coupled to" anything, but rather executed by a processor. The only sensible interpretation of "boot components" must include something broader than software alone. Thus, Apple's proposal of "module" is more appropriate than Rembrandt's proposal of "one or more software layers." Rembrandt attempts to confuse the issue by citing to usage of the phrases "bootstrap code" and "software components" in the specification and the prosecution history, but it is inappropriate to assume that those phrases were intended to have the same meaning as "boot component." Indeed, it is more reasonable to assume that those phrases were intended to mean something *other than* "boot component." *See generally CAE Screenplates, Inc. v. Heinrich Fiedler GmbH & Co. KG*, 224 F.3d 1308, 1317 (Fed. Cir. 2000) ("In the absence of any evidence to the contrary, we must presume that the use of these different terms in the claims connotes different meanings.").

With respect to the parties' dispute as to whether boot components are used by the system BIOS, the language of claim 1 clearly indicates that they are: "a plurality of boot components coupled to said expansion bus and ***accessed by said processor when said system BIOS is executed***." '678 Patent at col. 22:4–6 (emphasis added). Rembrandt argues that the kernel operating system is a boot component that is not used by the system BIOS (ECF No. 76 at 11), but again, Rembrandt is incorrect. First, this argument is inconsistent with the plain language of claim 1, which states that boot components are accessed while the system BIOS is executed. Furthermore, the system BIOS uses the operating system kernel when it loads and passes control to the operating system kernel. '678 Patent at col. 8:12–31. (*See also* Buscaino Decl. ¶ 3.) Even Rembrandt acknowledges this in its proposed construction of "system BIOS": "one or more layers of software that perform startup operations, including initializing hardware and ***transferring control to a layer that loads the operating system***." (ECF No. 76 at 11 (emphasis added).)

E. "system BIOS"

Terms & Claims	Apple's Construction	Rembrandt's Construction
"system BIOS" (Claims 1 and 4)	"firmware that controls the computer system from startup until the transfer of control to a boot component that loads the operating system kernel"	"one or more layers of software that perform startup operations, including initializing hardware and transferring control to a layer that loads the operating system kernel"

The dispute regarding this term relates to whether the "system BIOS" is more appropriately described as "firmware" or "one or more layers of software." The term "firmware" is narrower than the term "software"; firmware is a specific type of software that is installed firmly in hardware and cannot easily be modified. (Buscaino Decl. ¶ 4.) The use of the term "firmware" in relation to the '678 Patent appropriately highlights the fact that the system BIOS is not easily modifiable. The '678 Patent itself defines firmware to mean "system BIOS or expansion cards" in describing the prior art. '678 Patent at col. 2:22–23 Although this definition is provided in the context of describing the prior art, that does not diminish its importance, particularly because it is provided in the context of describing something that the prior art failed to do: "[Clark] does not address the verification of any firmware (system BIOS or expansion cards)." *Id.* Thus, the applicants were attributing their own definition of firmware to a system BIOS, not a prior art definition. The file history also refers to "ROM software" as "firmware." (Scarsi Decl. ¶ 3, Ex. B at 7.)

In addition, the portion of the '678 Patent that discusses recovery differentiates between system BIOS firmware and other software or configuration data stored on a boot component. The '678 Patent contemplates recovery of BIOS failures through a "shadowing" process in which the system BIOS firmware resident on the ROM BIOS is not replaced when a failure is detected; instead, firmware resident on the AEGIS ROM is used to boot the system. '678 Patent at col. 10:56–61. On the other hand, other software or configuration data may actually be replaced if they fail. '678 Patent at col. 10:61–67. In this way, failures of the BIOS firmware are differentiated from other failures in which the failed software or configuration data—which

is stored less firmly in the hardware than the system BIOS—may actually be replaced rather than merely shadowed.

F. "a host computer"

Terms & Claims	Apple's Construction	Rembrandt's Construction
"a host computer" (Claim 3)	"a separate computer system connected to the computer through a communications interface"	"a server computer attached to a network"

The primary difference between the parties' proposed constructions (in light of Rembrandt's revised construction[5]) is Rembrandt's use of the word "server" before computer. While the '678 Patent refers to certain host computers as servers, that does not imply that the host computer must be a server computer. A server implies that the network host responds to commands from a client or workstation. The '678 Patent is directed towards a computer system (*e.g.* col. 1:22–24) and is not exclusively directed towards a client computer or workstation. Therefore it is inappropriate to import the server limitation. Rembrandt's construction also suffers, because there is not clarity as to what is referred to by a "server computer."[6]

Apple's proposed construction is well supported. It is similar to the language of claim 3, which requires that the communication link between the separate host computer and the computer at issue be made "through a communications interface." (*See also* Scarsi Decl. ¶¶ 4-5,

[5] Rembrandt's opening claim construction brief (ECF No. 76) proposes a different construction than Rembrandt included in the joint claim construction statement (ECF No. 72-1), in which proposed that "a host computer" should be construed as "a computer, attached to a network, providing primary services such as computation, data base access or special programs or programming languages."

[6] Microsoft Dictionary, 3rd edition defines server as:

> 1. On a local area network (LAN), a computer running administrative software that controls access to the network and its resources, such as printers and disk drives, and provides resources to computers functioning as ***workstations*** on the network.
>
> 2. On the Internet or other network, a computer or program that responds to commands from a ***client***. For example, a file server may contain an archive of data or program files; when a client submits a request for a file, the server transfers a copy of the file to the client.

(Scarsi Decl. ¶ 5, Ex. D at 8 (emphases added).)

Ex. C and Ex. D (both defining host as "The main computer in a system of computers or terminals connected by communications links.").)

G. "Power on Self Test (POST)"

Terms & Claims	Apple's Construction	Rembrandt's Construction
"Power on Self Test (POST)" (Claim 4)	"one or more diagnostic tests performed by the system BIOS upon system startup"	"one or more diagnostic tests performed by the computer system during its startup"

The parties dispute relates to whether the "Power on Self Test (POST)" is "performed by the system BIOS." Intrinsic and extrinsic evidence demonstrate that the POST as contemplated in the '678 Patent is performed by the system BIOS. Figure 1b shows the BIOS performing POST at step 152. Even though Figures 1b and 2b–2d show POST being initiated before the BIOS is activated, the BIOS is immediately activated after POST initialization and subsequently performs the POST checks. As the specification notes, "[a]ll of these tests, except for the initial processor self test[7], are under the control of system BIOS 112." '678 Patent at col. 8:10–11; *see also* col. 8:12–13 ("Once system BIOS 112 has performed all of its power on tests . . . .").

The '678 Patent explains four ways to invoke POST and refers to "First layer 110 includes system BIOS 112 and corresponds to the first phase of the bootstrap process. The first phase of the boot process is the Power on Self Test or POST." '678 Patent at col. 7:61–63.

Moreover, a widely used BIOS boot specification published around the time of the alleged conception of the invention describes the POST as "the part of the BIOS that takes control immediately after the computer is turned on." (Scarsi Decl. ¶ 6, Ex. E.) For the foregoing reasons, it is appropriate to link the system BIOS and the POST in the construction of POST, as in Apple's proposed construction.

[7] The processor self test is an internal self test performed by the processor itself, distinct from the POST. (Buscaino Decl. ¶ 8.)

H. "when said boot component fails, recovering said boot component" and "to replace said boot component"

Terms & Claims	Apple's Construction	Rembrandt's Construction
"when said boot component fails, recovering said failed boot component. . . . to replace said failed boot component." (Claims 4 and 7)	"at the time boot component fails its integrity verification, automatically recovering said failed boot component . . . automatically replacing said failed boot component."	"when said boot component fails its integrity verification, recovering said boot component . . . to replace said boot component that failed its integrity verification"

1. "when said boot component fails, recovering said boot component"

Following Rembrandt's brief, Apple addresses this term in three subparts. Rembrandt describes the dispute as follows: "The parties dispute (i) whether 'when' means 'at the time'; (ii) whether this claim term requires 'automatically recovering'; and (iii) whether a 'boot component' or a 'failed boot component' is recovered."

Regarding issue (i), Apple agrees with Rembrandt that "when" may have different meanings depending on the context. *See, e.g.*, *Renishaw PLC v. Marposs Societa Per Azioni*, 158 F.3d 1243, 1251 (Fed. Cir. 1998) (observing that "when" can mean "at or during the time that; just at the moment that; at any or every time that; [or] at, during, or after the time that"). Here, when means "at the time" because the '678 Patent discloses an automated verification and recovery procedure in which the recovery process is entered as soon as an integrity failure is detected. The annotated version of Figure 2b provided by Rembrandt (reproduced below) supports Apple's construction, not Rembrandt's. Rembrandt argues that because integrity checking (*e.g.*, step 286) and the recovery process (step 298) are distinct steps, they are not occuring simultaneously. This is true, and is consistent with Apple's proposed construction. Apple uses "at the time" synonoumously with "as soon as" to convey that once an integrity failure is detected, the recovery process is entered immediately. This construction is supported

by Figure 2b, which shows a line directly connecting an integrity verification step (*e.g.*, 286) and the recovery process (step 298), which is to be followed when an integrity failure is detected.

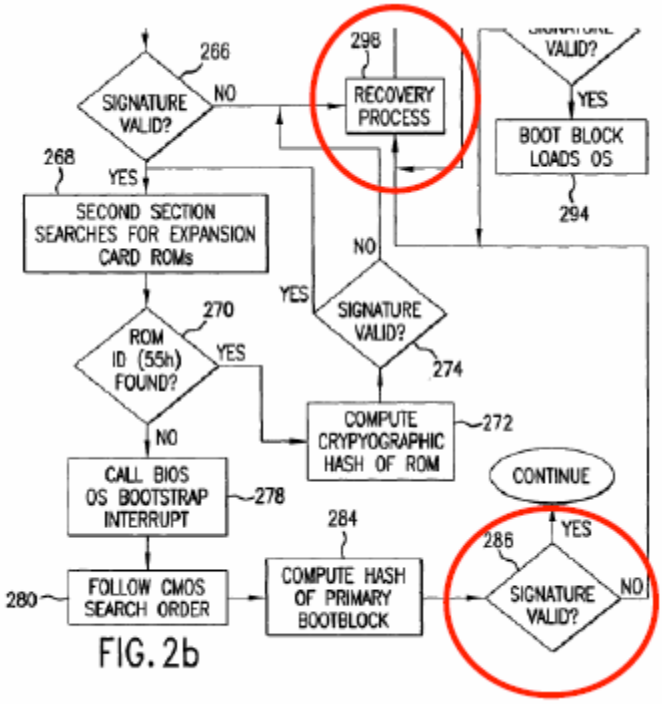

FIG. 2b

Regarding issue (ii), for the reasons explained above, the '678 Patent relates only to automated recovery and should be construed to that effect. *See, e.g.*, '678 Patent at col. 6:25 ("This entire process occurs without user intervention."); col. 10:8 (same sentence).

Regarding issue (iii), Apple does not believe there is a dispute between the parties. Apple is not proposing a construction in which a "failed boot component" is replaced with another failed boot component as Rembrandt attempts to argue. Apple's construction merely states that the failed boot component is automatically recovered and replaced upon failure of the integrity verification.

2. "to replace said boot component"

For the reasons discussed above, it is appropriate to include the word "automatically" in the construction of this phrase.

I. "secure protocol"

Terms & Claims	Apple's Construction	Rembrandt's Construction
"secure protocol" (Claim 7)	"a standard enabling communication between the computer system and the trusted repository secured through a cryptographic algorithm"	"cryptography combined with a protocol to add security to the recovery process"

Apple's proposed construction is well supported by the evidence. The patent describes "a secure protocol [used] to inform a trusted repository that a failure has occurred and to obtain a valid replacement component." '678 Patent at col. 4:49–51. Therefore, Apple describes the secure protocol in the context of the Patent as "enabling communication between the computer system and the trusted repository."

Rembrandt takes issue with Apple's construction of "protocol" as "a standard enabling communication." Rembrandt opts instead to not construe the word "protocol" at all. This is not helpful and will not assist the jury. *See Power-One, Inc. v. Artesyn Techs., Inc.*, 599 F.3d 1343, 1348 (Fed. Cir. 2010) ("The terms, as construed by the court, must ensure that the jury fully understands the court's claim construction rulings and what the patentee covered by the claims.") (internal citation and quotation marks omitted). Apple's construction of "protocol" comports with the specification and relevant extrinsic evidence. The specification states that "the present invention can be based on well known networking protocols . . . or on a custom protocol or various combinations of known protocols." '678 Patent at col. 4:51–56. Two versions of Microsoft's Computer Dictionary define "communications protocol" as a "set of rules or standards designed to enable computers to connect with one another and to exchange information with as little error as possible. . . ." (Scarsi Decl. ¶¶ 4-5, Ex. C; Ex. D.)

Rembrandt is critical in particular of Apple's use of the word "standard," arguing that its use precludes the use of custom protocols, which are contemplated by the Patent. (*See* ECF No. 76 at 26–27.) Rembrandt's argument misses the point. Regardless of whether a protocol is

custom or not, it must use a standard procedure or system of rules that all participants understand in order to effectuate communication. (Buscaino Decl. ¶ 7.)

The parties agree that the protocol is made secure through cryptography. Apple proposes that the Court adopt the phrase "cryptographic algorithm" in relation to cryptography to mirror the language of the specification. *See, e.g.*, '678 Patent at col. at 4:56–57 ("Cryptographic algorithms are combined with the chosen protocols to add security to the recovery process. . . .").

Apple's proposed construction should be adopted because it is more closely tied to the specification and construes the entire term, not just one of two words.

J. "coupled to"

Terms & Claims	Apple's Construction	Rembrandt's Construction
"coupled to" (Claims 1 and 3)	"directly linked to"	"connected directly or indirectly"

The intrinsic evidence makes clear that "coupled to" as used in the '678 Patent means "directly linked to." The phrase appears six times in the claims of the patent and nowhere else. In each of the six instances, the phrase is used to mean "directly linked to." The first three instances of "coupled to" appear in claim 1 and describe the expansion bus directly linked to (1) a "processor"; (2) a "memory storing a system BIOS"; and (3) a "plurality of boot components." '678 Patent at col. 21:42 –22:1–6. Figure 1c of the '678 Patent makes clear that in each of these three instances "coupled to" describes a direct link. Indeed, Figure 1c shows the bus (#6) directly linked to, *inter alia*, the processor (#4), the ROM BIOS (#2) (i.e., memory storing a system BIOS), and boot components (*e.g.*, #12, a hard disk drive, and #14, the communications interface).

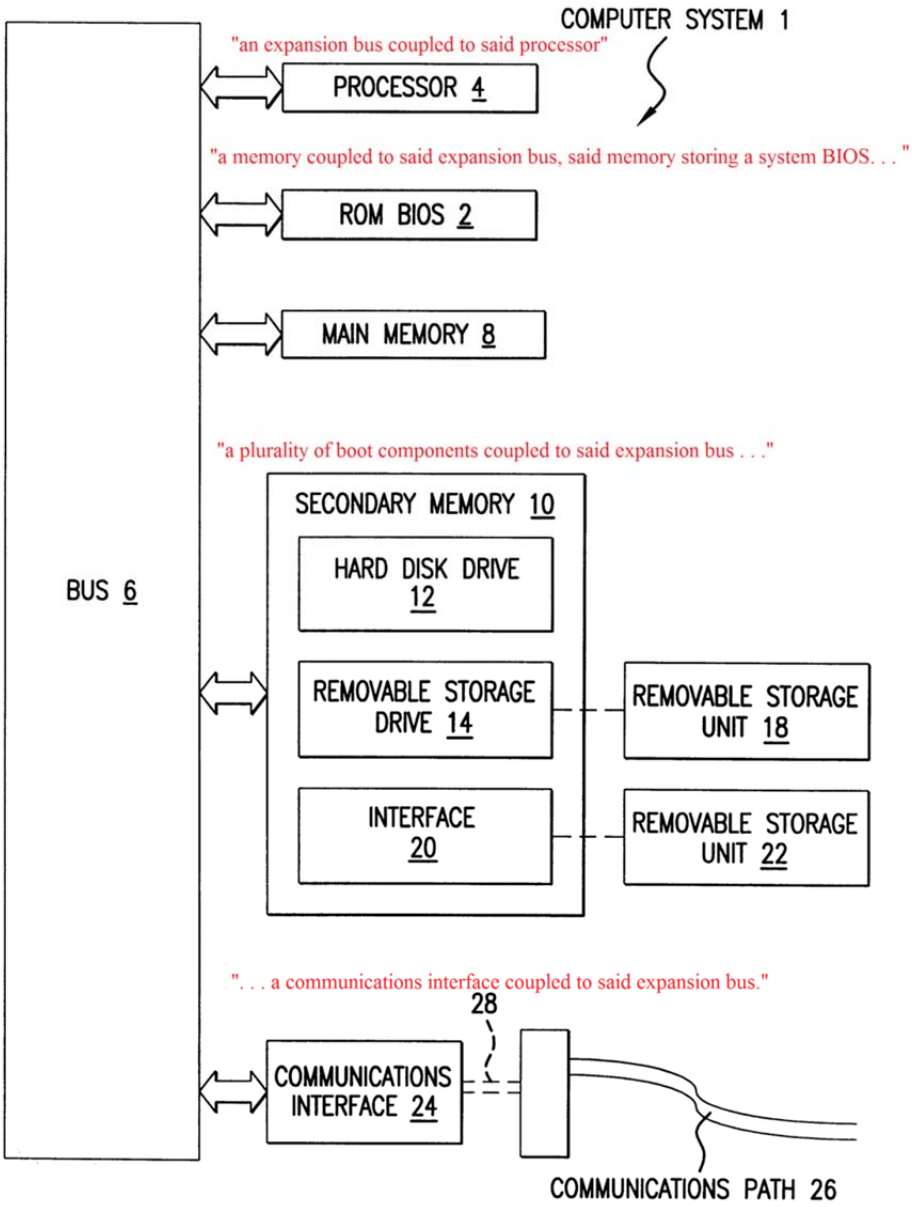

FIG.1c

Similarly, claim 3 describes the connection between a communications interface and the expansion bus: "[a]n architecture for initializing a computer system according to claim 1, wherein said trusted repository is a host computer communicating with said computer system through a communications interface coupled to said expansion bus." '678 Patent at col. 22:15–19. As shown in Figure 1c, that connection is direct.

The remaining two instances of "coupled to" relate to the "trusted repository" and similarly describe a direct link. First, claim 2[8] describes "an expansion ROM coupled to said expansion bus." The expansion ROM would be plugged into the computer system and thus directly connected to its bus. (Buscaino Decl. ¶ 6.) Additionally, the '678 Patent describes an exemplary expansion ROM as a hard disk card ('678 Patent at col. 8:12–16), which would also plug directly into a bus. The last use of the "coupled to" language appears in the portion of Claim 1 relating to trusted repositories: "a trusted repository coupled to said expansion bus." '678 Patent at col. 22:7. Rembrandt reads into this language a "communications interface provid[ing] an indirect connection between the trusted repository and the expansion bus" to support its construction. (ECF No. 76 at 7.) But no such requirement is found in the Patent. Instead, this language should be read in the context of the five other uses of "coupled to" in the '678 Patent, all of which are in the context of a direct connection. Claims 2 and 3 are particularly instructive in that they describe the expansion ROM (an exemplary trusted repository) and the communications interface—both of which would be directly connected to the bus—as "coupled to" the bus. (Claim 3 notably does not state that the host computer is "coupled to" anything.) The '678 Patent thus does not require a communications interface to serve as the indirect connection between a trusted repository and the expansion bus. Furthermore, the '678 Patent contemplates the AEGIS ROM to contain a trusted repository ('678 Patent at col. 10:56–64), and the AEGIS ROM is part of the system BIOS (#2), which is directly linked to expansion bus, as shown in Figure 2a. Because the AEGIS ROM always contains a replacement copy of the system BIOS, there is always a trusted repository on the AEGIS ROM that is directly connected the expansion bus. Hence, the language of claim 1 describing "a trusted repository coupled to said expansion bus" requires no indirect connection.

The cases Rembrandt cites that interpret "coupled to" broadly are inapposite. In those cases, the intrinsic evidence required that "coupled to" be interpreted broadly. *See, e.g.,*

[8] Rembrandt does not allege infringement of claim 2.

Negotiated Data Solutions, LLC v. Dell, Inc., 596 F. Supp. 2d 949, 964 (E.D. Tex. 2009). ("The claims, when read in light of the specification, indicate that 'coupled' means more than a 'direct physical connection.'"). Here, the opposite is true—the claims, when read in light of the specification (Figure 1c in particular), indicate that "directly connected to" is the appropriate construction. The instant case is more analogous to *PCTEL, Inc. v. Agere Systems*, in which, after a review of the claim language and intrinsic evidence, the Court found that "the clearest reading of 'a device coupled to the local bus' is a 'device directly connected to the local bus'." No. C03-02474, 2006 U.S. Dist. LEXIS 25943, at *19 (N.D. Cal. Mar. 20, 2006). Accordingly, Apple's construction of "coupled to" should be adopted.

V. CONCLUSION

For the foregoing reasons, Apple respectfully requests that the Court adopt Apple's proposed constructions, which comport with the intrinsic and extrinsic evidence and the law.

Dated: October 8, 2014

Respectfully submitted,

By: /s/ Mark C. Scarsi

Mark C. Scarsi (*admitted Pro Hac Vice*)
mscarsi@mibank.com
Miguel Ruiz (*admitted Pro Hac Vice*)
mruiz@milbank.com
Ashlee N. Lin (*admitted Pro Hac Vice*)
ashlee.lin@milbank.com
MILBANK, TWEED, HADLEY &MCCLOY LLP
601 South Figueroa Street, 30th Floor
Los Angeles, California 90017-5735
Telephone: (213) 892-4000
Facsimile: (213) 629-5063

Christopher J. Gaspar (*admitted Pro Hac Vice*)
cgaspar@milbank.com
Andrew Lichtenberg (*admitted Pro Hac Vice*)
alichtenberg@milbank.com
MILBANK, TWEED, HADLEY & MCCLOY LLP
1 Chase Manhattan Plaza
New York, New York 10005
Telephone: (212) 530-5000
Facsimile: (212) 530-5219

Melissa Smith (Bar No. 24001351)
melissa@gillamsmithlaw.com
GILLAM & SMITH L.L.P
303 S. Washington Ave
Marshall, Texas 75670
Telephone: (903) 934-8450
Facsimile: (903) 934-9257

Attorneys for Defendant Apple Inc.

CERTIFICATE OF SERVICE

I certify that counsel of record who are deemed to have consented to electronic service are being served on October 8, 2014, with a copy of this document via the Court's CM/ECF systems per Local Rule CV-5(a)(3). Any other counsel will be served electronic mail, facsimile, overnight delivery and/or First Class Mail on this date.

\s\ Mark C. Scarsi
Mark C. Scarsi